1. Application is highly available to the user.

To achieve high application availability on AWS, leverage redundancy through multi-AZ deployments, automated scaling with Auto Scaling, and load balancing with ELB, ensuring minimal downtime and resilient application performance.

Key AWS Services & Concepts for High Availability:

* **Availability Zones (AZs):**

AWS Regions consist of multiple isolated data centers (AZs) with redundant power, networking, and cooling, enabling you to deploy applications across AZs for resilience against failures within a single AZ.

* **Amazon EC2 Auto Scaling:**

This service automatically adjusts the number of EC2 instances based on demand, ensuring your application can handle fluctuations in traffic and maintain performance.

* **Elastic Load Balancing (ELB):**

ELB distributes incoming traffic across multiple instances, ensuring that traffic is evenly distributed and that no single instance becomes a point of failure.

* **Amazon Relational Database Service (RDS) Multi-AZ Deployments:**

For databases managed by RDS, enable Multi-AZ deployments to replicate the data across AZs, ensuring that the database remains available even if one AZ becomes unavailable.

* **Amazon Simple Storage Service (S3):**

As an object storage service, S3 offers high availability and durability, making it suitable for storing static content, backups, and other data assets.

* **AWS Lambda:**

You can run code without provisioning or managing servers using AWS Lambda, allowing you to scale your applications to handle spikes in traffic without worrying about server infrastructure.

* **Amazon CloudFront:**

This content delivery network (CDN) caches and distributes content globally, providing low-latency access to users worldwide and enhancing application availability.

* **Architectural Pattern:**

Design your application to have no single point of failure, and consider redundancy through multiple instances, failover mechanisms, and automated monitoring.

+-------------------+

| Clients |

+-----+--------------+

|

+----|----| |

| | | |

+--+ +-- +--+

| | | |

+->| | | +--|

| | | | |

+->|ELB|->|Auto Scaling|

| | | | |

| +--| | | +

| | | | |

+------> | | | | <------ +

|EC2 | |

| | |

+---+--- | | |

| | |

| S3(Static) +-------+ |

| | | | |

+--------+-------------| <-----

|

|

+----------+

| | |

+--------+

RDS Multi-AZ (Database)

|

+----------------+

| Application |

+----------------+

Explanation of the Diagram:

1. **Clients:** Users access the application.
2. **ELB:** Traffic is distributed across multiple application instances.
3. **Auto Scaling:** The system automatically scales up or down to handle load.
4. **EC2 Instances:** Web servers or application servers hosted on EC2.
5. **S3(Static):** Stores static assets like images, CSS, and Javascript.
6. **RDS Multi-AZ:** Stores the database in a multi-AZ configuration.
7. **Application:** The core application logic.

2) Application should be performance excellence.

3) Application is highly secured.

4) Cost optimized